

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平9-55718

(43) 公開日 平成9年(1997)2月25日

(51) Int. Cl. ⁵	識別記号	序内整理番号	F I	技術表示箇所
H 0 4 L	1/00		H 0 4 L	1/00
	12/56	9466-5K		11/20
	29/08			13/00
				E
				1 0 2 F
				3 0 7 Z

審査請求 有 請求項の数 6 O L (全 13 頁)

(21) 出願番号 特願平7-208946

(22) 出願日 平成7年(1995)8月17日

(71) 出願人 000004237

日本電気株式会社

東京都港区芝五丁目7番1号

(72) 発明者 上島 良之

東京都港区芝五丁目7番1号 日本電気株式会社内

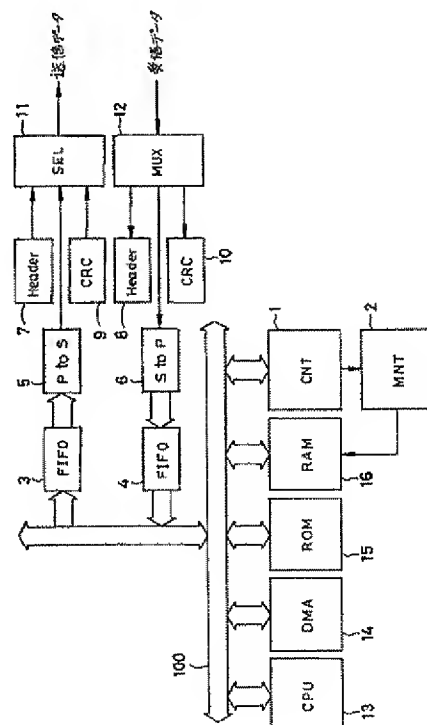
(74) 代理人 弁理士 ▲柳▼川 信

(54) 【発明の名称】 データ通信装置

(57) 【要約】

【課題】 通信状態がよければパケットのデータ長を長くして通信速度を向上させ、通信状態が悪ければパケットのデータ長を短くしてエラー状態からの回復を早くする。

【解決手段】 RAM 16 には送信すべきデータ及び受信したデータが蓄積される。RAM 16 内の送信データは送信データ長カウンタ 1 に通信状態に応じて書込まれた送信データ長にしたがってバッファ管理回路 2 で管理される。バッファ管理回路 2 が管理している RAM 16 内の送信データは DMA 14 によってデータバス 100 を介して送信 FIFO 3 に転送され、パラレルシリアル変換回路 5 とセクタ 11 とを介して受信側に送信される。



【特許請求の範囲】

【請求項1】 1つのファイルを複数のパケットに分割して通信する時に通信先の装置からの応答を確認せずにパケットを送信するデータ通信装置であって、前記通信先の装置との間の通信状態に応じて前記パケットのデータ長を可変する可変手段を有することを特徴とするデータ通信装置。

【請求項2】 前記可変手段は、前記通信先の装置からの肯定応答を受信した時にパケットデータ長を長くしかつ前記通信先の装置からの否定応答を受信した時にパケットデータ長を短くするよう構成したことを特徴とする請求項1記載のデータ通信装置。

【請求項3】 受信データ及び送信データを蓄積しておくためのランダムアクセスメモリを含み、前記ランダムアクセスメモリに蓄積された前記送信データを複数のパケットに分割して通信する時に通信先の装置からの応答を確認せずにパケットを送信するデータ通信装置であって、前記通信先の装置との間の通信状態に応じて前記パケットのデータ長を可変する可変手段を有することを特徴とするデータ通信装置。

【請求項4】 前記可変手段は、前記通信先の装置からの肯定応答を受信した時に前記ランダムアクセスメモリからデータ長が長いパケットを読み出しかつ前記通信先の装置からの否定応答を受信した時に前記ランダムアクセスメモリからデータ長が短いパケットを読み出すよう構成したことを特徴とする請求項3記載のデータ通信装置。

【請求項5】 受信データ及び送信データを蓄積しておくためのランダムアクセスメモリと、データ通信上のデータ通信速度を調整するために前記送信データを一時記憶するための一時記憶手段と、前記ランダムアクセスメモリと前記一時記憶手段との間の送信データのデータ転送を制御する制御手段とを含み、前記ランダムアクセスメモリに蓄積された前記送信データを複数のパケットに分割して通信する時に通信先の装置からの応答を確認せずにパケットを送信するデータ通信装置であって、前記通信先の装置との間の通信状態に応じて前記パケットのデータ長を可変する可変手段を有することを特徴とするデータ通信装置。

【請求項6】 前記可変手段は、前記通信先の装置からの肯定応答を受信した時に前記ランダムアクセスメモリと前記一時記憶手段との間でデータ長が長いパケットをデータ転送するよう前記制御手段に指示しかつ前記通信先の装置からの否定応答を受信した時に前記ランダムアクセスメモリと前記一時記憶手段との間でデータ長が短いパケットをデータ転送するよう前記制御手段に指示するよう構成したことを特徴とする請求項5記載のデータ通信装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明はデータ通信装置に関

し、特に送信側と受信側との間で予め設定された通信手順（通信プロトコル）にしたがってパケットデータの送受信を行うデータ通信装置に関する。

【0002】

【従来の技術】 従来、この種のデータ通信装置においては、例えば端末のコンピュータがホストのコンピュータからデータをダウンロードする場合等、コンピュータ間でデータを送受信する時に用いられる。

【0003】 この場合、コンピュータはデータを数百から数千バイト程度の大きさに分割し、分割したデータにヘッダやエラーチェック用の情報を付加してフレームを構成してパケット化している。このパケット化したデータをコンピュータ相互間で取り決めた通信プロトコルにしたがって送受信している。

【0004】 通信プロトコルとしては、すでに幾つか決められたものがある。例えば、図13～図15に示すように、送信側がパケットデータを送信した後に、そのパケットデータの正常受信または異常受信を示す受信側からの応答を待つて次のパケットデータの送信または同じパケットデータの再送を行うものがある。この種の通信プロトコルとしては、例えば、XMODEMやYMODEMがある。

【0005】 以下、図13～図15を用いてその通信プロトコルについて説明する。まず、送信側では、図13に示すように、パケットデータ①の送信処理を実行すると（図13ステップS61）、そのパケットデータ①の送信処理に対する受信側からの応答の待ち状態となる（図13ステップS62）。

【0006】 送信側では受信側からの応答パケットを受信すると、その応答パケットの受信処理を行い（図13ステップS63）、その応答パケットがACK（Acknowledge）（肯定応答）パケットか、あるいはNACK（Negative Acknowledge）（否定応答）パケットかを判断する（図13ステップS64）。

【0007】 送信側では例えばパケットデータ②を送信した時に受信側からNACKパケットを受信すると、ステップS61に戻って同じパケットデータ②の再送を行う。また、送信側では受信側からACKパケットを受信すると、全てのパケットデータが送信完了したかを判定する（図13ステップS65）。

【0008】 送信側は全てのパケットデータが送信完了したと判定すると処理を終了し、送信完了していないと判定すると次の送信パケットデータの処理を実行する（図13ステップS66）。

【0009】 一方、受信側では、図14に示すように、送信側からパケットデータ①、②が送信されてくると、そのパケットデータ①、②の受信処理を行い（図14ステップS71）、受信したパケットデータ①、②を解析してデータ長の確認やチェックサムまたはCRC（Cy

clic Redundancy Check code) 等によるエラーチェックを行って受信パケットがエラーか否かを判定する(図14ステップS72)。

【0010】受信側は受信パケットにエラーがないと判定すると、正しくデータを受信することができたということを送信側に伝えるためにACKパケットを送信する(図14ステップS73)。また、受信側は受信パケットにエラーがあると判定すると、正しくデータを受信することができなかったということを送信側に伝えるためにNACKパケットを送信する(図14ステップS74)。上記の処理動作のシーケンスは図15に示す通りである。

【0011】この場合、送信側では受信側からのNACKパケットを受信するか、あるいはACKもNACKも受信できなかった場合に前回送信したパケットデータ②とおなじパケットデータ②を再送する。また、これらの通信プロトコルではパケットデータの長さが固定となっている。

【0012】上記の通信プロトコル以外に、送信側が受信側からの応答を待たずにパケットデータを順次送信し、受信側で異常が検出された時にその異常を示す応答を、予め設定された数だけ正常受信した時にその正常受信を示す応答を夫々受信側から送信側に返送するものもある。この種の通信プロトコルとしては、例えば、ZMODEMやB-P ius、及びQuick-VAN等がある。

【0013】以下、図16～図18を用いてその通信プロトコルについて説明する。まず、送信側では、図16に示すように、パケットデータの送信か受信かを判断し(図16ステップS81)、パケットデータの送信と判断すると、パケットデータの送信処理を全てのパケットデータの送信が完了するまで実行する(図16ステップS81～S84)。すなわち、送信側では受信側からのACKの返送を待たずに、パケットデータ①～⑨を次々に送信するようになっている。

【0014】送信側はパケットデータの受信と判断すると、パケットデータの受信処理を行い(図16ステップS85)、受信側からの応答パケットがACKパケットかあるいはNACKパケットかを判断する(図16ステップS86)。

【0015】送信側では受信側からの応答パケットがACKパケットであると判断するとパケットデータの送信処理をそのまま続行し、応答パケットがNACKパケットであると判断するとそのNACKパケットに付加されたパケット番号のパケットデータ④を再送する(図16ステップS87)。

【0016】送信側では受信側からNACKパケットまたはACKパケットの返送を受信した時、既に幾つかのパケットデータの送信が終了しているの、NACKに対応するパケットデータ④以降の送信も完了しているこ

とになる。しかしながら、送信側がパケットデータ①～⑨を送信する時にパケットデータ①～⑨各々にパケット番号を付加して送信しているの、受信側ではどのパケットデータでエラーが発生したのかをパケット番号で通知すれば、送信側がそのパケット番号のパケットデータ④だけを再送することが可能となる。

【0017】一方、受信側では、図17に示すように、送信側からパケットデータ①～⑨が送信されてくると、そのパケットデータ①～⑨の受信処理を行い(図17ステップS91)、受信したパケットデータを解析してデータ長の確認やチェックサムまたはCRC等によるエラーチェックを行って受信パケットがエラーか否かを判定する(図17ステップS92)。

【0018】受信側は受信パケットにエラーがないと判定すると、正しくデータを受信することができた回数が予め設定されたN回(Nは正の整数)となったか否かを判定する(図17ステップS93)。受信側は正しくデータを受信することができた回数がN回になると、正しくデータを受信したことを送信側に伝えるためにACKパケットを送信する(図17ステップS94)。

【0019】また、受信側は受信パケットにエラーがあると判定すると、正しくデータを受信できなかったことを送信側に伝えるためにNACKパケットを送信する(図17ステップS95)。上記の処理動作のシーケンスは図18に示す通りである。

【0020】図12は従来のデータ通信装置の構成例を示すブロック図である。図において、従来のデータ通信装置はCPU51によって制御され、CPU(中央処理装置)51を動作させるためのプログラムはROM(リードオンリメモリ)53に格納されている。

【0021】RAM(ランダムアクセスメモリ)54には送信すべきデータ及び受信したデータが蓄積される。RAM54内の送信データはDMA(ダイレクトメモリアクセス)52によってデータバス300を介して送信FIFO(ファーストイン・ファーストアウト)41に転送される。この場合、RAM54内の送信データはDMA52を用いることなく、CPU51がRAM54から読出して送信FIFO41に書込む方法も可能である。

【0022】まず、送信時には送信FIFO41に書込まれた送信データがパラレルシリアル変換回路(P l o S)43によってパラレル形式のデータからシリアル形式のデータに変換される。

【0023】CPU51がパケットの送信開始を指示すると、データ部の送信に先だってパケットの先頭を示すデータ、送信相手を指定するアドレス、パケットの順番を示す番号等を含むヘッダがヘッダ発生回路(Header)45から送信される。続いて、パラレルシリアル変換回路43でシリアル形式のデータに変換されたデータ部がセレクト(SE L)49で選択されて送信され

る。

【0024】設定されたデータ長分のデータが送信し終わると、DMA52またはCPU51はRAM54から送信FIFO41へのデータの転送を終了する。続いて、セレクト49が切替えられ、通信相手側でのデータエラーチェック用としてCRCデータまたはチェックサムデータがCRC発生回路(CRC)47から送信され、パケットデータの送信が終了する。

【0025】一方、受信時には受信したデータがマルチプレクサ(MUX)50によってヘッダ部とデータ部とエラーチェック部とに分離され、ヘッダ部はヘッダ解析回路(Header)46で処理され、エラーチェック部はCRC照合回路(CRC)48で処理される。

【0026】データ部に対しては送信時の処理と全く逆の処理が行われる。つまり、データ部はシリアルパラレル変換回路(S to P)44でシリアル形式のデータからパラレル形式のデータに変換される。

【0027】シリアルパラレル変換回路44でパラレル形式のデータに変換された受信データは受信FIFO42に順次書込まれていく。受信FIFO42に書込まれた受信データはDMA52によって受信FIFO42からRAM54に転送される。この場合も、送信時と同様に、CPU51が介在して受信データを受信FIFO42からRAM54に転送させることも可能である。

【0028】上述した如く、ROM53にはCPU51を動作させるためのプログラムが格納されており、そのプログラムにしたがって通信プロトコルが実現される。上記の通信プロトコルの処理手順は図13及び図14と図16及び図17とに夫々示す通りである。

【0029】図13及び図14に示す通信プロトコルの場合、送信側では送信処理によって1つのパケットを送信すると、受信側からの応答の待ち状態となる。受信側からの応答がなく、タイムアウトした場合にはそのパケットデータの再送を行い、受信側からの応答があった場合には受信処理によって受信側からの応答パケットを受信する。

【0030】送信側では受信した応答パケットを解析し、ACKパケットかあるいはNACKパケットかを判断する。NACKパケットを受信した場合には、送信したパケットデータと同じパケットデータの再送を行う。ACKパケットを受信した場合には、全てのパケットデータの送信が終了したかどうかを判断し、全てのパケットデータの送信が終了すれば、データ通信処理を終了する。全てのパケットデータの送信が終了していなければ、次のパケットデータを送信する。

【0031】次に、受信側ではパケットデータを受信すると、そのパケットデータにエラーがあるかどうかを判断し、エラーがあった場合にはNACKパケットを送信し、エラーがなければACKパケットを送信する。

【0032】図16及び図17に示す通信プロトコルの

場合、送信側ではCPUに対する割込みの発生によって送信処理か受信処理かの判断を行う。送信側ではパケットデータの送信要求が発生したならばパケットの送信処理を行い、パケットデータの受信を検出したならばパケットデータの受信処理を行う。

【0033】送信側では受信したパケットデータの種類が判断され、NACKパケットを受信したならば、そのNACKパケットで指示されたパケットデータを再送する。送信側ではACKパケットを受信したならば、次の処理に移行する。送信処理あるいは受信処理を行うと、送信側では全てのパケットデータの送信が完了したかどうかを判断し、全てのパケットデータの送信が完了するとデータ通信処理を終了し、まだ送信すべきパケットデータがあれば次のパケットデータの送信処理を行う。

【0034】受信側ではパケットデータの受信処理を行うと、その受信したパケットデータにエラーがあるかどうかを判断する。このとき、エラーの有無の判断にはCRC照合回路からの情報やデータ長が正しいかどうか等を考慮して行う。エラーが発生している場合にはNACKパケットを送信する。

【0035】また、エラーが発生していない場合には、N回受信する毎にACKパケットを送信するようにすると、パケットデータをN回受信したかどうかを判断し、パケットデータをN回受信する毎にACKパケット送信処理を行い、N回でなければ処理を終了する。

【0036】

【発明が解決しようとする課題】一般的に、パケットデータ通信では実際に送りたいパケットデータにアドレスやパケット番号、及びエラーチェックのための情報等が付加されるので、パケットデータ長が短ければ実際のデータに対して付加される部分のデータの割合が大きくなってしまい、その分データ通信中にエラーが発生する確率が高くなり、付加されたデータの分だけ通信効率が悪くなる。逆に、パケットデータ長が長ければ、データ通信中にエラーが発生して再送する必要が生じた場合、再送しなければならぬデータ量も大きくなる。

【0037】上述した従来のデータ通信装置では、送信側がパケットデータを送信した後に、そのパケットデータの正常受信または異常受信を示す受信側からの応答を待って次のパケットデータの送信または同じパケットデータの再送を行う通信プロトコルの場合、パケットデータ送信毎にACKパケットの確認を必ず必要としているので、全てのデータの転送完了までに時間がかかり、通信効率が悪くなる。

【0038】また、この通信プロトコルの場合にはパケットデータ長が固定であるため、回線の状況がよくても必ず実際のデータ以外にヘッダやCRCが付加されるので、通信効率が悪くなる。さらに、エラーの発生確率が悪くなり、再送時のデータ量も大きくなる等の問題がある。

【0039】一方、送信側が受信側からの応答を待たずにパケットデータを次々に送信する通信プロトコルの場合、送信側ではACKパケットの返送で確認されたパケットデータ以外は受信側に送信し終わってもバッファしておく必要があり、バッファ用のメモリが必要となる。

【0040】また、送信側及び受信側ともにエラーがあった場合には前のデータに戻らなくてはならないので、バッファ内のパケットデータの順番の管理が複雑になる。仮に、バッファの管理を軽減するために、エラーのあったパケットデータ以降を全て再送するようにすると、再送するデータが多くなり、通信効率が悪くなる。

【0041】特開平4-243343号公報には、変復調装置の復調側に設けた復調エラーベクトル出力装置から出力される復調エラーベクトルの大きさを基に信号伝播路の状態を判断し、信号伝播路の状態の判断結果に基づいてデータパケット長決定装置によって変復調装置の送信データのパケット長の最適化処理を行ってデータ通信効率を向上させるパケット通信方式が開示されている。

【0042】この方式では送信データのパケット長の最適化を行っているのであるが、自分で受信している状況から信号伝播路の状態を判断し、自分の送信するパケット長を決定しているため、送信方向と受信方向とで信号伝播路の状況が異なる場合には判断を誤ることがある。

【0043】また、送信方向と受信方向とで通信するデータ量が異なる場合、例えば上記のような通信プロトコルを用いてデータ転送を行う場合には、データパケットを送信する側に比べてACKパケットまたはNACKパケットを送信する側では通信するデータ量が少ない。したがって、このパケット通信方式では信号伝播路の状態を判断する情報量も少ないため、適切な信号伝播路の状態を判断することが困難である。

【0044】また、特開平2-125551号公報には、複数のデータパケットを通信回線を通じて順次受信側に伝送するにあたり、1個のデータパケットを伝送する毎に、受信されたデータパケットのエラー状態に応じて受信側から発せられる再送要求の有無を検出し、再送要求があれば再送要求の対象とされたデータパケットを再度受信側に伝送するとともに、その再送回数を計数し、再送要求がなければ次に伝送すべき新たなデータパケットについてのパケット長を、新たなデータパケットの直前のデータパケットについて計数された再送回数に応じて設定することで、データ通信効率を向上させるデータパケット伝送方法が開示されている。

【0045】この方法では受信側から発せられる再送要求の回数から判断してデータパケット長を設定しているが、送信側においては1個のデータパケットを伝送する毎に、受信側から発せられる再送要求を検出しているため、データ通信効率が悪くなる。

【0046】そこで、本発明の目的は上記の問題点を解

消し、通信状態がよければパケットのデータ長を長くして通信速度を向上させることができ、通信状態が悪ければパケットのデータ長を短くしてエラー状態からの回復を早くすることができるデータ通信装置を提供することにある。

【0047】

【課題を解決するための手段】本発明によるデータ通信装置は、1つのファイルを複数のパケットに分割して通信する時に通信先の装置からの応答を確認せずにパケットを送信するデータ通信装置であって、前記通信先の装置との間の通信状態に応じて前記パケットのデータ長を可変する可変手段を備えている。

【0048】本発明による他のデータ通信装置は、受信データ及び送信データを蓄積しておくためのランダムアクセスメモリを含み、前記ランダムアクセスメモリに蓄積された前記送信データを複数のパケットに分割して通信する時に通信先の装置からの応答を確認せずにパケットを送信するデータ通信装置であって、前記通信先の装置との間の通信状態に応じて前記パケットのデータ長を可変する可変手段を備えている。

【0049】本発明による別のデータ通信装置は、受信データ及び送信データを蓄積しておくためのランダムアクセスメモリと、データ通信上のデータ通信速度を調整するために前記送信データを一時記憶するための一時記憶手段と、前記ランダムアクセスメモリと前記一時記憶手段との間の送信データのデータ転送を制御する制御手段とを含み、前記ランダムアクセスメモリに蓄積された前記送信データを複数のパケットに分割して通信する時に通信先の装置からの応答を確認せずにパケットを送信するデータ通信装置であって、前記通信先の装置との間の通信状態に応じて前記パケットのデータ長を可変する可変手段を備えている。

【0050】

【発明の実施の形態】まず、本発明の作用について以下に述べる。

【0051】1つのファイルを複数のパケットに分割して通信する時に通信先の装置からの応答を確認せずにパケットを送信するデータ通信装置において、通信先の装置との間の通信状態、つまり受信側からのACKパケットまたはNACKパケットの受信に応じてパケットのデータ長を可変する。

【0052】これによって、通信状態がよければパケットのデータ長を長くして通信速度を向上させることが可能となり、通信状態が悪ければパケットのデータ長を短くしてエラー状態からの回復を早くすることが可能となる。

【0053】すなわち、通信状態がよい時にパケットのデータ長を長くすると、全体のパケット数が少なくなり、ヘッダやCRC等のオーバーヘッド部分を少なくすることができるので、通信時間全体で見ると従来の通信時

10

20

30

40

50

間よりも短縮することが可能となる。

【0054】また、通信状態が悪い時にパケットのデータ長を短くすると、エラーが発生した場合でも再送するデータ量を少なくすることができるので、通信時間全体で見ると従来の通信時間よりも短縮することが可能となる。

【0055】次に、本発明の実施例について図面を参照して説明する。図1は本発明の一実施例の構成を示すブロック図である。図において、本発明の一実施例によるデータ通信装置は送信データ長カウンタ(CNT)1と、バッファ管理回路(MNT)2と、送信FIFO(ファーストイン・ファーストアウト)3と、受信FIFO4と、パラレルシリアル変換回路(P to S)5と、シリアルパラレル変換回路(S to P)6と、ヘッダ発生回路(Header)7と、ヘッダ解析回路(Header)8と、CRC発生回路(CRC)9と、CRC照合回路(CRC)10と、セクタ(SEL)11と、マルチプレクサ(MUX)12と、CPU(中央処理装置)13と、DMA(ダイレクトメモリアクセス)14と、ROM(リードオンリメモリ)15と、RAM(ランダムアクセスメモリ)16とから構成されている。

【0056】尚、送信データ長カウンタ1と、送信FIFO3と、受信FIFO4と、CPU13と、DMA14と、ROM15と、RAM16とはデータバス100を介して互いに接続されている。

【0057】本発明の一実施例によるデータ通信装置はCPU13によって制御され、CPU13を動作させるためのプログラムはROM15に格納されている。RAM16には送信すべきデータ及び受信したデータが蓄積される。RAM16内の送信データはDMA14によってデータバス100を介して送信FIFO3に転送される。この場合、RAM15内の送信データはDMA14を用いることなく、CPU13がRAM15から読出して送信FIFO3に書込む方法も可能である。

【0058】まず、送信時には送信FIFO3に書込まれた送信データがパラレルシリアル変換回路5によってパラレル形式のデータからシリアル形式のデータに変換される。

【0059】CPU13がパケットの送信開始を指示すると、データ部の送信に先だってパケットの先頭を示すデータ、送信相手を指定するアドレス、パケットの順番を示す番号等を含むヘッダがヘッダ発生回路7から送信される。続いて、パラレルシリアル変換回路5でシリアル形式のデータに変換されたデータ部がセクタ11で選択されて送信される。

【0060】設定されたデータ長分のデータが送信し終わると、DMA14またはCPU13はRAM15から送信FIFO3へのデータの転送を終了する。続いて、セクタ11が切替えられ、通信相手側でのデータエラ

ーチェック用としてCRCデータまたはチェックサムデータがCRC発生回路9から送信され、パケットデータの送信が終了する。

【0061】このとき、CPU13から送信データ長カウンタ1に通信状態に応じた送信データ長が書込まれる。バッファ管理回路2は送信データ長カウンタ1に書込まれた値にしたがってRAM16内のデータを管理し、RAM16内のどの部分をパケット化するかを決定するとともに、そのパケット長を決定する。

【0062】一方、受信時には受信したデータがマルチプレクサ12によってヘッダ部とデータ部とエラーチェック部とに分離され、ヘッダ部はヘッダ解析回路8で処理され、エラーチェック部はCRC照合回路10で処理される。

【0063】データ部に対しては送信時の処理と全く逆の処理が行われる。つまり、データ部はシリアルパラレル変換回路6でシリアル形式のデータからパラレル形式のデータに変換される。

【0064】シリアルパラレル変換回路6でパラレル形式のデータに変換された受信データは受信FIFO4に順次書込まれていく。受信FIFO4に書込まれた受信データはDMA14によって受信FIFO4からRAM15に転送される。この場合も、送信時と同様に、CPU13が介在して受信データを受信FIFO4からRAM15に転送させることも可能である。

【0065】図2は本発明の一実施例によるパケット送信処理を示すフローチャートであり、図3は本発明の一実施例によるパケット受信処理を示すフローチャートであり、図4は本発明の一実施例の処理動作を示すシーケンスチャートである。これら図1～図4を用いて本発明の一実施例の処理動作について説明する。

【0066】送信側ではCPU13に対する割込みの発生によって送信処理か受信処理かの判断を行う(図2ステップS1)。送信側ではパケットデータの送信要求が発生したならばパケットの送信処理を行い(図2ステップS2)、パケットデータの受信を検出したならばパケットデータの受信処理を行う(図2ステップS5)。

【0067】送信側では受信したパケットデータの種類が判断され(図2ステップS6)、NACKパケットを受信したならば、CPU13が以前の送信データ長よりもデータ長を減少させて送信データ長カウンタ1に指示し(図2ステップS8)、そのNACKパケットで指示されたパケットデータを再送する(図2ステップS9)。このとき、RAM16内の再送すべきパケットデータはバッファ管理回路2によってそのパケット長とともに制御される。

【0068】送信側ではACKパケットを受信したならば、CPU13が以前の送信データ長よりもデータ長を増加させて送信データ長カウンタ1に指示し(図2ステップS7)、次の処理に移行する。

【0069】送信処理あるいは受信処理を行うと、送信側では全てのパケットデータの送信が完了したかどうかを判断し（図2ステップS3）、全てのパケットデータの送信が完了するとデータ通信処理を終了し、まだ送信すべきパケットデータがあれば次のパケットデータの送信処理を行う（図2ステップS4）。

【0070】受信側ではパケットデータの受信処理を行うと（図3ステップS11）、CPU13がCRC照合回路10の照合結果を基にその受信したパケットデータにエラーがあるかどうかを判断する（図3ステップS12）。このとき、CPU13によるエラーの有無の判断にはCRC照合回路10からの情報やデータ長が正しいかどうか等を考慮して行う。エラーが発生している場合にはNACKパケットを送信する（図3ステップS15）。

【0071】また、エラーが発生していない場合にはN回受信する毎にACKパケットを送信するように構成されているとすると、CPU13はパケットデータをN回受信したかどうかを判断し（図3ステップS13）、パケットデータをN回受信する毎にACKパケット送信処理を行い（図2ステップS14）、N回でなければ処理を終了する。

【0072】例えば、図4に示すように、送信側から受信側に次々と送信するパケットデータ①～⑧のうち、受信側がパケットデータ②を受信した時にN回受信が検出されたとすると、受信側から送信側にACKパケットが送信される。送信側ではそのACKパケットを受信すると、ACKパケットを受信してから次に送信されるパケットデータ⑤以降の送信データ長を増加させる。

【0073】これに対し、送信側から受信側に次々と送信するパケットデータ①～⑧のうち、受信側がパケットデータ④を受信した時にエラーを検出したとすると、受信側から送信側にNACKパケットが送信される。

【0074】送信側ではそのNACKパケットで指示されたパケットデータ④を再送するが、このとき送信データ長が減少されているので、パケットデータ④-1、④-2として受信側に送信する。それ以降、送信側から受信側には減少された送信データ長でパケットデータ④-1、④-2、⑦、⑧が送信される。

【0075】図5は本発明の一実施例による処理動作で用いられる送信データ長の増減が自在なパケットデータ例を示す図である。図において、パケットデータの最後尾には最終フラグが付加されており、このパケットデータの受信時にその最終フラグを確認することで、送信データ長が増減されてもそのパケットデータの受信が可能となる。

【0076】図6は図5に示すパケットデータを用いる場合の送信処理を示すフローチャートであり、図7は図5に示すパケットデータを用いる場合の受信処理を示すフローチャートである。これら図5～図7を用いて送信

データ長を増減した時の送受信処理について説明する。

【0077】CPU13はパケットデータの送信処理を行う際に（図6ステップS21）、RAM16から送信FIFO3に転送されるデータの最終データの検出を行う（図6ステップS22）。

【0078】CPU13はRAM16から送信FIFO3に転送されるデータの最終データを検出すると、そのパケットデータに最終フラグを付加して送信するように制御する（図6ステップS23）。尚、本実施例は最終フラグをCPU13の制御によってCRC発生回路9で付加するよう構成されている。

【0079】一方、受信側においてDMA14またはCPU13はパケットデータの受信処理を行う際に（図7ステップS31）、受信FIFO4からRAM16に転送されるデータの最終フラグの検出を行う（図7ステップS32）。

【0080】CPU13は受信FIFO4からRAM16に転送されるデータの最終フラグを検出すると、そのパケットデータのエラーの有無を判断し（図7ステップS33）、エラーが検出されるとそのエラーに対する処理を実行する（図7ステップS34）。CPU13はそのパケットデータにエラーがないと判断すると、またはエラー処理が終了すると、パケットデータ受信処理を終了する。

【0081】図8は本発明の一実施例による処理動作で用いられる送信データ長の増減が自在なパケットデータの他の例を示す図である。図において、パケットデータの先頭にはそのパケットデータのデータ長が付加されており、このパケットデータの受信時にそのデータ長を確認することで、送信データ長が増減されてもそのパケットデータの受信が可能となる。

【0082】図9は図8に示すパケットデータを用いる場合の送信処理を示すフローチャートであり、図10は図8に示すパケットデータを用いる場合の受信処理を示すフローチャートである。これら図8～図10を用いて送信データ長を増減した時の送受信処理について説明する。

【0083】CPU13は予め設定したデータ長を受信側に送信し（図9ステップS41）、その後にパケットデータの送信処理を行う（図9ステップS42）。尚、本実施例ではデータ長をCPU13の制御によってヘッダ発生回路7から送信するよう構成されている。

【0084】CPU13はRAM16から送信FIFO3に転送するデータのデータ長が最初に送信したデータ長となるまで、そのパケットデータを送信するように制御する（図9ステップS42、S43）。

【0085】一方、受信側においてCPU13は送信側からデータ長を受信すると（図10ステップS51）、そのデータ長を保持する（データ長→A）（図10ステップS52）。

【0086】その後、CPU13はパケットデータの受信処理を行い(図10ステップS53)、受信FIFO4からRAM16に受信データを転送させる。その時、CPU13は送信側から受信したパケットデータのデータ長をカウントする(データ長→B)(図10ステップS54)。

【0087】CPU13は送信側から受信したデータ長(A)と送信側から受信したパケットデータのデータ長(B)とを比較し(図10ステップS55)、それらが一致するまでパケットデータの受信処理を行う(図10ステップS53~S55)。

【0088】CPU13はそれらの一致を検出すると、そのパケットデータのエラーの有無を判断し(図10ステップS56)、エラーが検出されるとそのエラーに対する処理を実行する(図10ステップS57)。CPU13はそのパケットデータにエラーがないと判断すると、またはエラー処理が終了すると、パケットデータ受信処理を終了する。

【0089】図11は本発明の他の実施例の構成を示すブロック図である。図において、本発明の他の実施例によるデータ通信装置は送信データ長カウンタ(CNT)21と、送信FIFO(ファーストイン・ファーストアウト)22と、受信FIFO23と、パラレルシリアル変換回路(P to S)24と、シリアルパラレル変換回路(S to P)25と、ヘッダ発生回路(Header)26と、ヘッダ解析回路(Header)27と、CRC発生回路(CRC)28と、CRC照合回路(CRC)29と、セクタ(SEL)30と、マルチプレクサ(MUX)31と、CPU(中央処理装置)32と、DMA(ダイレクトメモリアクセス)33と、ROM(リードオンリメモリ)34と、RAM(ランダムアクセスメモリ)35とから構成されている。

【0090】尚、送信データ長カウンタ21と、送信FIFO22と、受信FIFO23と、CPU32と、DMA33と、ROM34と、RAM35とはデータバス200を介して互いに接続されている。

【0091】本発明の他の実施例によるデータ通信装置はCPU32によって制御され、CPU32を動作させるためのプログラムはROM34に格納されている。RAM35には送信すべきデータ及び受信したデータが蓄積される。RAM35内の送信データはDMA33によってデータバス100を介して送信FIFO22に転送される。

【0092】まず、送信時には送信FIFO22に書込まれた送信データがパラレルシリアル変換回路24によってパラレル形式のデータからシリアル形式のデータに変換される。

【0093】CPU32がパケットの送信開始を指示すると、データ部の送信に先だってパケットの先頭を示すデータ、送信相手を指定するアドレス、パケットの順番

を示す番号等を含むヘッダがヘッダ発生回路26から送信される。続いて、パラレルシリアル変換回路24でシリアル形式のデータに変換されたデータ部がセクタ30で選択されて送信される。

【0094】設定されたデータ長分のデータが送信し終わると、DMA33またはCPU32はRAM35から送信FIFO22へのデータの転送を終了する。続いて、セクタ30が切替えられ、通信相手側でのデータエラーチェック用としてCRCデータまたはチェックサムデータがCRC発生回路28から送信され、パケットデータの送信が終了する。

【0095】このとき、CPU13から送信データ長カウンタ21に通信状態に応じた送信データ長が書込まれる。DMA33は送信データ長カウンタ1に書込まれた送信データ長分だけRAM35から送信FIFO22にデータを転送する。

【0096】一方、受信時には受信したデータがマルチプレクサ31によってヘッダ部とデータ部とエラーチェック部とに分離され、ヘッダ部はヘッダ解析回路27で処理され、エラーチェック部はCRC照合回路29で処理される。

【0097】データ部に対しては送信時の処理と全く逆の処理が行われる。つまり、データ部はシリアルパラレル変換回路25でシリアル形式のデータからパラレル形式のデータに変換される。

【0098】シリアルパラレル変換回路25でパラレル形式のデータに変換された受信データは受信FIFO23に順次書込まれていく。受信FIFO23に書込まれた受信データはDMA33によって受信FIFO23からRAM35に転送される。尚、本発明の他の実施例でも、図5~図7に示すように、パケットデータに最終フラグを付加するか、あるいは図8~図10に示すように、パケットデータにデータ長を付加することによってパケットデータのデータ長を可変自在としている。

【0099】このように、1つのファイルを複数のパケットに分割して通信する時に通信先の装置からの応答を確認せずにパケットを送信するデータ通信装置において、通信先の装置との間の通信状態、つまり受信側からのACKパケットまたはNACKパケットの受信に応じてパケットのデータ長を可変することによって、通信状態がよければパケットのデータ長を長くして通信速度を向上させることができ、通信状態が悪ければパケットのデータ長を短くしてエラー状態からの回復を早くすることができる。

【0100】すなわち、通信状態がよい時にパケットのデータ長を長くすると、全体のパケット数が少なくなり、ヘッダやCRC等のオーバーヘッド部分を少なくすることができるので、通信時間全体で見ると従来の通信時間よりも短縮することが可能となる。

【0101】また、通信状態が悪い時にパケットのデー

タ長を短くすると、エラーが発生した場合でも再送するデータ量を少なくすることができるので、通信時間全体で見ると従来の通信時間よりも短縮することが可能となる。

【0102】

【発明の効果】以上説明したように本発明によれば、1つのファイルを複数のパケットに分割して通信する時に通信先の装置からの応答を確認せずにパケットを送信するデータ通信装置において、通信先の装置との間の通信状態に応じてパケットのデータ長を変換することによって、通信状態がよければパケットのデータ長を長くして通信速度を向上させることができ、通信状態が悪ければパケットのデータ長を短くしてエラー状態からの回復を早くすることができるという効果がある。

【図面の簡単な説明】

【図1】本発明の一実施例の構成を示すブロック図である。

【図2】本発明の一実施例によるパケット送信処理を示すフローチャートである。

【図3】本発明の一実施例によるパケット受信処理を示すフローチャートである。

【図4】本発明の一実施例の処理動作を示すシーケンスチャートである。

【図5】本発明の一実施例による処理動作で用いられる送信データ長の増減が自在なパケットデータ例を示す図である。

【図6】図5に示すパケットデータを用いる場合の送信処理を示すフローチャートである。

【図7】図5に示すパケットデータを用いる場合の受信処理を示すフローチャートである。

*【図8】本発明の一実施例による処理動作で用いられる送信データ長の増減が自在なパケットデータの他の例を示す図である。

【図9】図8に示すパケットデータを用いる場合の送信処理を示すフローチャートである。

【図10】図8に示すパケットデータを用いる場合の受信処理を示すフローチャートである。

【図11】本発明の他の実施例の構成を示すブロック図である。

【図12】従来例の構成を示すブロック図である。

【図13】従来例によるパケット送信処理の一例を示すフローチャートである。

【図14】従来例によるパケット受信処理の一例を示すフローチャートである。

【図15】従来例の処理動作の一例を示すシーケンスチャートである。

【図16】従来例によるパケット送信処理の他の例を示すフローチャートである。

【図17】従来例によるパケット受信処理の他の例を示すフローチャートである。

【図18】従来例の処理動作の他の例を示すシーケンスチャートである。

【符号の説明】

1, 21 送信データ長カウンタ

2 バッファ管理回路

3, 22 送信FIFO

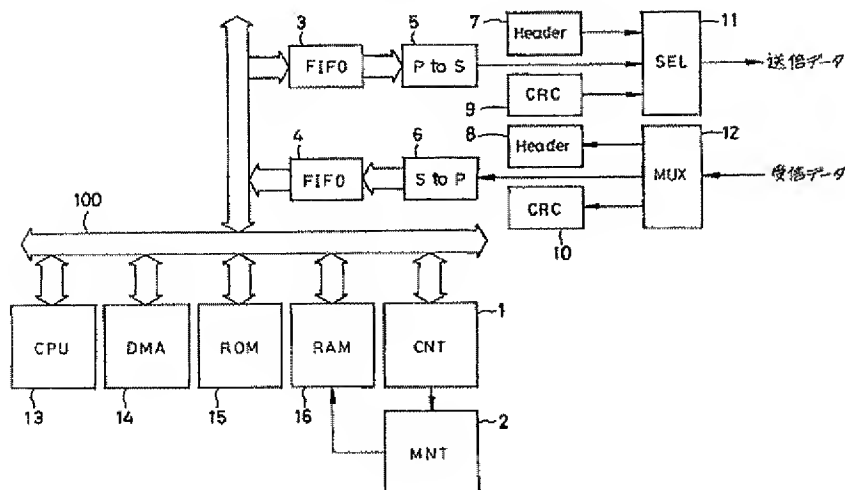
4, 23 受信FIFO

13, 32 CPU

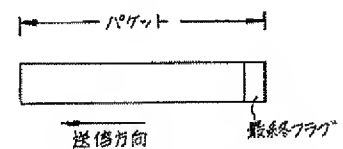
14, 33 DMA

*30 16, 35 RAM

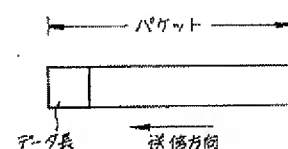
【図1】



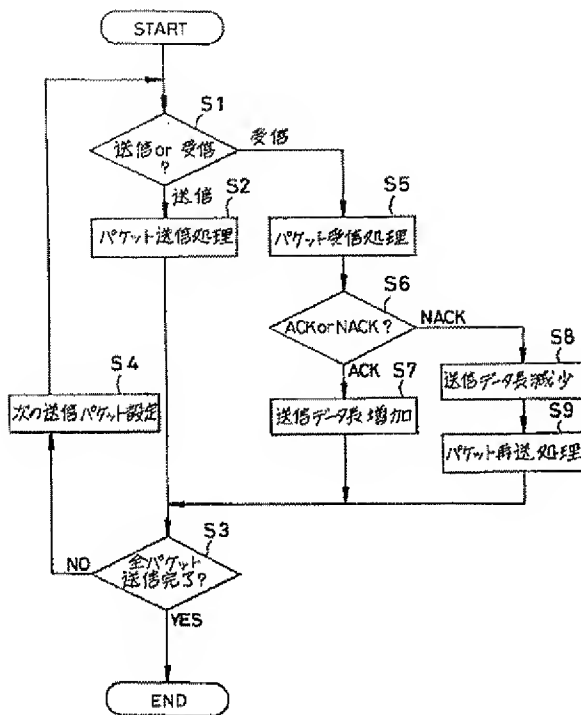
【図5】



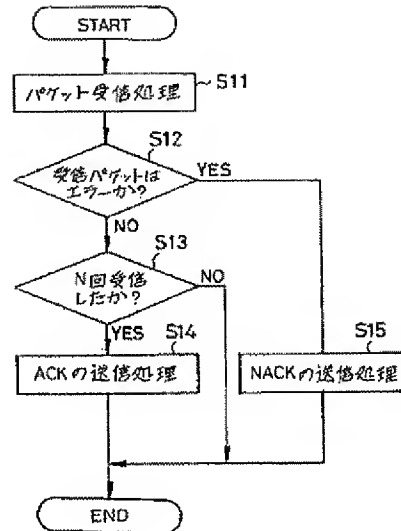
【図8】



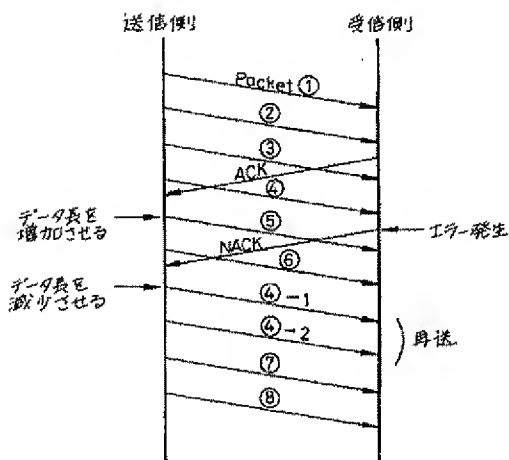
【図2】



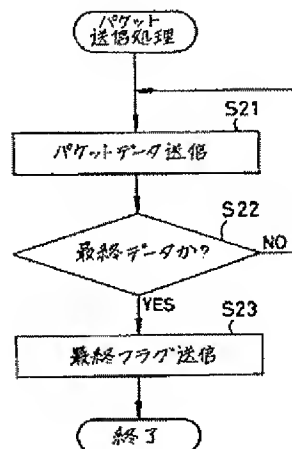
【図3】



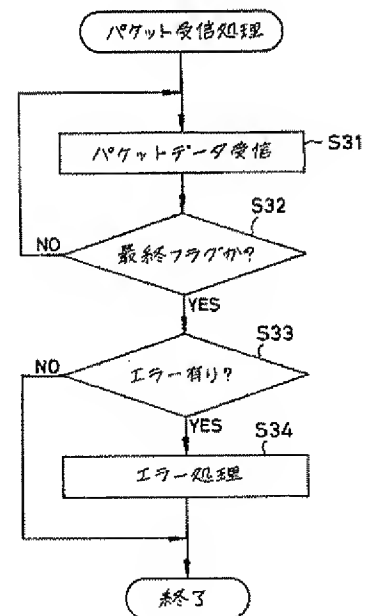
【図4】



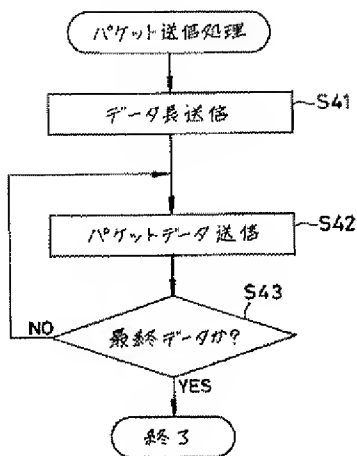
【図6】



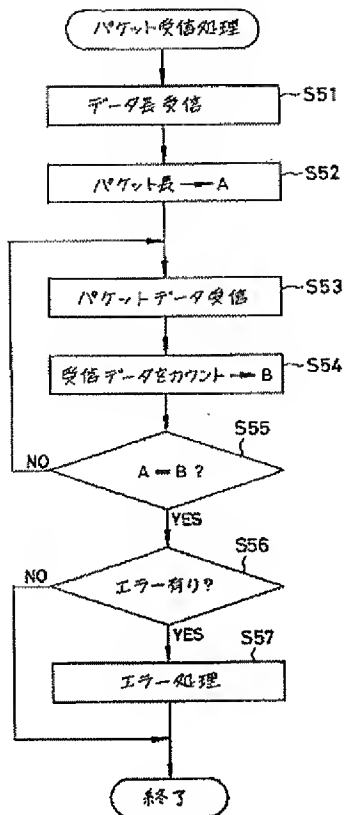
【図7】



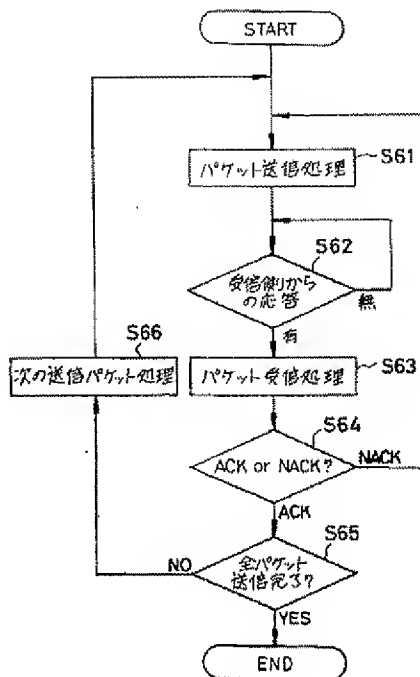
【図9】



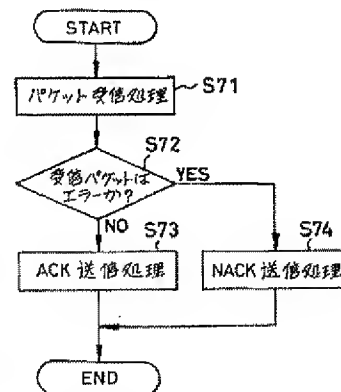
【図10】



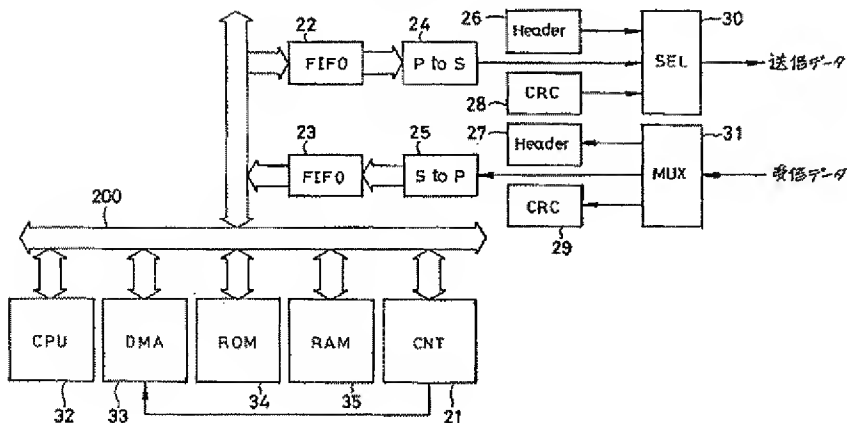
【図13】



【図14】



【図11】



```

sequenceDiagram
    participant S as 送信側!
    participant R as 受信側!
    S->>R: Packet ①
    R->>S: ACK
    S->>R: Packet ②
    R->>S: NACK
    S->>R: Packet ③
    R->>S: ACK
    S->>S: 再送
  
```

```

graph TD
    START([START]) --> S81{S81  
送信or  
受信?}
    S81 -- 送信 --> S82[パケット送信処理]
    S81 -- 受信 --> S85[パケット受信処理]
    S82 --> S83{S83  
全パケット  
送信完了?}
    S83 -- YES --> END([END])
    S83 -- NO --> S84[パケット再送信処理]
    S84 --> S81
    S85 --> S86{S86  
ACK or NACK?}
    S86 -- ACK --> S83
    S86 -- NACK --> S87[パケット再送信処理]
    S87 --> S81
  
```

The flowchart illustrates the packet transmission/reception process. It begins with a 'START' terminal, leading to decision point S81: '送信or 受信?' (Transmit or Receive?). If the action is '送信' (Transmit), it proceeds to S82: 'パケット送信処理' (Packet transmission processing). If the action is '受信' (Receive), it proceeds to S85: 'パケット受信処理' (Packet reception processing). From S82, the flow goes to decision point S83: '全パケット 送信完了?' (All packets transmission completed?). If 'YES', it ends at 'END'. If 'NO', it goes to S84: 'パケット再送信処理' (Packet retransmission processing) and loops back to S81. From S85, the flow goes to decision point S86: 'ACK or NACK?'. If 'ACK', it proceeds to S83. If 'NACK', it goes to S87: 'パケット再送信処理' (Packet retransmission processing) and loops back to S81.

```

graph TD
    START([START]) --> S91[パケット受信処理 S91]
    S91 --> S92{受信パケット  
はエラーか? S92}
    S92 -- YES --> S95[NACKの送信処理 S95]
    S92 -- NO --> S93{N回受信  
したか? S93}
    S93 -- YES --> S94[ACKの送信処理 S94]
    S93 -- NO --> S95
    S94 --> END([END])
    S95 --> S93
  
```

The flowchart illustrates the packet reception process (S91). It begins with a START terminal, leading to the packet reception process (S91). A decision is made at S92: "Is the received packet an error?". If YES, the process proceeds to S95: "NACK transmission processing". If NO, the process proceeds to S93: "Has it been received N times?". From S93, if YES, it proceeds to S94: "ACK transmission processing", which then leads to the END terminal. If NO, it proceeds to S95. Both S94 and S95 lead back to the decision point S93.

【図18】

